UNCLASSIFIED

AD NUMBER
AD806373
LIMITATION CHANGES
TO: **Approved for public release; distribution is unlimited.**
FROM: **Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; JAN 1967. Other requests shall be referred to Office of Naval Research, 875 North Randolph Street, Arlington, VA 22203-1995.**
AUTHORITY
AFOSR ltr, 23 Mar 1967

THIS PAGE IS UNCLASSIFIED

ANALYSIS AND DESIGN OF COMMUNICATION NETWORKS WITH MEMORY

S. L. HAKIMI

TECHNICAL REPORT 67-103

JANUARY 1967

Information-Processing And Control SYSTEMS LABORATORY

NORTHWESTERN UNIVERSITY
EVANSTON, ILLINOIS
60201

TECHNICAL REPORT 67-103

ANALYSIS AND DESIGN OF COMMUNICATION NETWORKS WITH MEMORY

S. L. Hakimi

Electrical Engineering Department

This work was supported in part by the U. S. Air Force Office of Scientific Research, Grant AF-AFOSR-98-65, and in part by the Joint Services Electronics Program (U. S. Army, U. S. Navy and U. S. Air Force) under Office of Naval Research Contract Number N00014-66-C0020-A03 (Identification Number NR 373-502/3-14-66 Electronics Branch).

January 1967

Information-Processing And Control
SYSTEMS LABORATORY

Northwestern University

Evanston, Illinois

60201

TABLE OF CONTENTS

Analysis and Design of Communication
Networks with Memory

by

S. L. Hakimi *

ABSTRACT

A mathematical formulation of the communication networks with memory is presented assuming that the sources of traffic are deterministic but not necessarily time invariant. The formulations leads to a linear programming problem. Some generalizations and justifications of the choice of the model are discussed. The same basic formulation can be used as a tool for analysis as well as least-cost design or improvement of an existing network. Design of the memory systems and its relation with messages with priorities is considered.

Similar concepts are used to arrive at an approximate linear programming formulation of "street traffic".

This work was supported in part by the U.S. Air Force Office of Scientific Research, Grant AF-AFOSR-98-65, and in part by the Joint Services Electronics Program (U.S. Army, U.S. Navy, and U.S. Air Force) under Office of Naval Research Contract Number N00014-66-A0020-A03 (Identification Number NR 373-502/3-14-66 Electronics Branch).

* Department of Electrical Engineering, Northwestern University, Evanston, Illinois

I. Introduction

A communication or traffic network N is a set of n points (switching stations, cities) that are connected by m lines (channels, transmission lines, highways). A convenient model of such a system is a graph G with n vertices (or nodes), $v_1, v_2, \ldots, v_n$ and m branches $b_1, b_2, \ldots, b_m$. Each branch b_i is connected between a pair of distinct vertices v_k and v_j. The vertices and the branches in G correspond to the stations and lines in network N. We assume that each line in N allows traffic only in one direction, (or N is unilateral). (As it will be shown, this assumption is not a necessary part of the formulation.) Arrowheads are placed on corresponding branches of G to signify the allowable direction of traffic flow. With each branch b_i $(1 \le i \le m)$, there is associated a real non-negative number b_i^* which represents the capacity of corresponding line in N. The capacity of a branch in G is the maximum allowable rate of flow in the corresponding branch in N (in the direction of arrowhead). If the stations in N (or vertices in G) have no capacity restrictions or "delay characteristics", then, they merely represent junctions and such a network is called a "circuit switching network" or a communication network with direct traffic [1,2,3] . Traffic handling capability and the design of such networks have been widely studied [4-10] . Now, let us attach a real non-negative number m_i to each vertex v_i $(i = 1,2,\ldots n)$ in G where m_i represents the maximum amount of messages (or traffic) that can be stored (or held up) at the corresponding station of N. In a communication network the number m_i is the

capacity of the memory system (or the capacity of the warehouse) at v_i . The function of the memory system (or unit) at a vertex is to store the messages in the memory unit when the outgoing lines from that vertex are busy (or flooded to the capacity). Such a network is called a network with "message and circuit switching" or a store-and-forward communication network [1,3] . We will refer to them as a communication network with memory. Although some engineering aspects of this type of networks have been studied, no general method for the analysis and design of such networks is available. A description of a general formulation for the communication networks with memory is the main result of this paper. This formulation leads to an extremely large linear program; however, this linear program has a special structure which enables one to take advantage of Danzig-Wolfe [11,12] decomposition principle to, partially, alleviate the computational difficulty. Also, it is shown that one can obtain a feasible solution to the design problem by a direct method which leads to suboptimal results.

II. Flow Model Formulation

Consider a communication network N represented by a weighted directed graph G with n vertices and m branches. The weight of the branch b_i is b_i^* $(i = 1,2,\ldots m)$ which represents the capacity of the corresponding line (channel) in N, and the weight of the vertex v_i is m_i $(i = 1,2,\ldots n)$ which represents the capacity of memory system located at the corresponding station in N. We construct a directed graph G^* from G as follows: We split each vertex v_i in G into two vertices v_{i1} and v_{i2} in G^*. All of the branches whose arrowheads are toward vertex v_i in G are connected to vertex v_{i1}

in G* and all of the branches whose arrowheads are away from v_i in G are connected to v_{i2} in G*. The branches in G* are labelled as the corresponding branches were in G. Let us assume that the message (or traffic) flow rate at the time t due to the messages originating from v_{i1} in branch b_j in G (or G*) is represented by $z_{ij}(t)$ ($1 \le j \le m$ and $1 \le i \le n$). Let the sum of the corresponding flow rates entering vertex v_j in G (or v_{j1} in G*) at time t be $x_{ij}(t)$ and the sum of flows leaving v_j in G (or v_{j2} in G*) at time t be $y_{ij}(t)$ ($1 \le i$ and $j \le n$). Figure 1(a) represents typical vertex v_i of G and Fig. 1(b) represents the splitting of corresponding vertex in G*. If there is no memory at v_j (i.e., $m_j = 0$), then $x_{ij}(t) \equiv y_{ij}(t)$ for all t. However if $m_j > 0$, then at some interval of time $x_{ij}(t) > y_{ij}(t)$ when the surplus incoming flows are accumulated in memory, and in another interval $x_{ij}(t) < y_{ij}(t)$ when the content of the memory is decreased.

We first consider the situation when all messages originate at one vertex, say v_{11}, called the "source." (This case corresponds to the "single-commodity" problem in network flow theory [7, 10]). The destinations of these messages could be at any or all vertices $v_{12}, v_{22}, \ldots v_{n2}$, called the "sinks." Let $s_1(t)$ be the rate of the production of messages at the source which is connected to v_{11} in G^* and let $u_{11}(t), u_{12}(t), \ldots, u_{1n}(t)$ be the rates of the arrival of messages at the sinks or the vertices $v_{12}, v_{22}, \ldots v_{2n}$, respectively*.

* Usually one assumes that each source produces messages that are destined for a single sink [7, 10]; thus one might have up to n sources connected to a single vertex. For the purpose of design, there is no reason to make such a distinction. However, if we are also to find a routing policy, then we must distinquish between the sources connected to a single vertex.

(Usually $s_1(t)$ and $u_{11}(t)$, $u_{12}(t)$, ..., $u_{1n}(t)$ are random processes; here, we might consider these functions to be expected values of the corresponding random variables at each instant of time). We refer to the source whose output is $s_1(t)$ as the source s_1.

At each vertex v_{j1} (or v_{j2}) of G* ($1 \leq j \leq n$), one may write a linear algebraic equation which expresses that the sum of flows entering v_{j1} (or leaving v_{j2}) due to the source s_1 is equal to $x_j(t)$ (or $y_j(t)$). In the matrix form, we may write for all t

$$A_1 \begin{bmatrix} \underline{Z}_1(t) \\ \\ s_1(t) \end{bmatrix} = \underline{X}_1(t) \text{ and } \qquad A_2 \begin{bmatrix} \underline{Z}_1(t) \\ \\ \underline{U}_1(t) \end{bmatrix} = \underline{Y}_1(t) \tag{1}$$

where column vectors $\underline{Z}_1(t) = [z_{11}(t)\ z_{12}(t)\ \ldots\ z_{1m}(t)]'$, $\underline{X}_1(t) = [x_{11}(t) x_{12}(t) \ldots\ x_{1n}(t)]'$, $\underline{Y}_1(t) = [y_{11}(t)\ y_{12}(t)\ \ldots\ y_{1n}(t)]'$, and $\underline{U}_1(t) = [u_{11}(t)\ u_{12}(t)\ \ldots\ u_{1n}(t)]'$. A_1 and A_2 are "zero and one" matrices and their form is self explanatory. In addition to the system of equations (1), we need some inequality constraints concering the capacities of the lines and memory units. The first system of inequalities is an obvious one.

$$0 \leq \underline{Z}_1(t) \leq \underline{B}^* \text{ for all } t, \text{where } \underline{B}^* = (b_1^*\ b_2^*\ \ldots\ b_m^*)' \tag{2}$$

That is, the capacity of each link limits the flow through the link. To arrive at the second system of inequalities, let us assume at time $t = 0$, r_{1i} units of messages due to s_1 are stored in m_i ($1 \leq i \leq n$). Let column vector $\underline{R}_1 = (r_{11} r_{12} \ldots r_{1n})'$ be called the initial state vector. We note that the function of memory was to store the surplus messages at each vertex. With this in mind, we may write for each instant of time t

$$\underline{0} \leq \underline{R}_1 + \int_0^t [\underline{X}_1(\tau) - \underline{Y}_1(\tau)]\, d\tau \leq \underline{M} \tag{3}$$

where $\underline{M} = (m_1 m_2 \ldots m_n)'$ and $\underline{0}$ is a zero column vector. The above inequalities

express that at no instant of time the accumulated sum of message stored in a vertex can exceed the capacity of that vertex nor can it be negative.

Before we discuss the formulation of the objective function, we generalize (1)-(3) to the case of many sources (or many commodities). Suppose to each vertex v_i in G (or v_{i1} in G^*) a source of messages is connected whose output as a function of time is $s_i(t)$. Corresponding to this source, we are given a sink vector $\underline{U}_i(t)$ whose components represent the rate of arrival of messages originating from the source connected to v_i at the different vertices of G. Note that we also need to have for each source s_i a separate initial state vector $\underline{R}_i$. Then, for each single source we may write a set of constraints in the form of (1) as follows

$$A_1 \begin{bmatrix} \underline{Z}_i(t) \\ \\ s_i(t) \end{bmatrix} = \underline{X}_i(t) \text{ and } \quad A_2 \begin{bmatrix} \underline{Z}_i(t) \\ \\ \underline{U}_i(t) \end{bmatrix} = \underline{Y}_i(t) \tag{4}$$

for $i = 1,2, \ldots, n$ and all t, where $\underline{Z}_i(t) = [z_{i1} - - - z_{im}]'$, $\underline{X}_i = [x_{i1} --- x_{in}]'$, and $\underline{Y}_i = [y_{i1} --- y_{in}]'$. Constraint (2) becomes

$$\sum_{i=1}^{n} \underline{Z}_i(t) \le \underline{B}^* \quad , \quad \underline{Z}_i(t) \ge \underline{0} \text{ for } i = 1,2, \ldots, n \text{ and all } t \tag{5}$$

and finally the constraints given by (3) become

$$\sum_{i=1}^{n} \left\{ \underline{R}_i + \int_0^t [\underline{X}_i(\tau) - \underline{Y}_i(\tau)]\, d\tau \right\} \le \underline{M}$$

and

$$\underline{R}_i + \int_0^t [\underline{X}_i(\tau) - \underline{Y}_i(\tau)]\, d\tau \ge \underline{0}, \text{ for } i = 1,2,\ldots n \text{ and all } t. \tag{6}$$

In a (linear) programming problem (4), (5), and (6) represent constraints and to complete the formulation we need an objective

function. However, to do this, we must specify exactly what the problem is? We define three separate problems and show that each problem leads to a different objective function, but the constraints remain unaltered.

Problem 1. Optimum Design: We are given G (the desired structure of the network) and the source output functions $s_i(t)$, a sink vector $\underline{U}_i(t)$, and initial state vectors $\underline{R}_i$ for $i = 1,2, \ldots, n$ and all t. We are also given two column vectors $\underline{C} = (c_1, c_2, \ldots c_m)'$ and $\underline{K} = (k_1, k_2, \ldots k_n)'$ where c_i and k_i are the cost per unit capacity of branch b_i and the memory unit m_i, respectively, $i = 1,2, \ldots n$ (or m). We would like to find the branch capacity vector $\underline{B}^*$ and the memory capacity vector $\underline{M}$ such that flow requirements are satisfied in the finite interval $[0, T]$ and overall cost is minimum. This problem may be expressed as follows: find $\underline{B}^*$ and $\underline{M}$ subject to constraints (4) through (6) for all t in the interval $[0, T]$ such that

$$\underline{C}^t \underline{B}^* + \underline{K}^t \underline{M} \tag{7}$$

is minimized. It should be noted that, for this problem to be solvable the following condition must be satisfied.

$$\int_0^T s_i(t)dt + \sum_{j=1}^{n} r_{ij} \geq \sum_{j=1}^{n} \int_0^T u_{ij}(t)\, dt \quad \text{for } i = 1,2,\ldots, n. \tag{8}$$

where r_{ij} and $u_{ij}(t)$ are the jth components of $\underline{R}_i$ and $\underline{U}_i(t)$.

Problem 2. Improvement of an Existing Network: The problem is to make a minimum cost modification of the present system, to make it capable of handling heavier traffic. Here G and present capacities $\underline{B}^*_p$ and $\underline{M}_p$ and initial state vectors $\underline{R}_i$ are given $(1 \leq i \leq n)$. In addition we are given a desired flow specifications $s_i(t)$ and $\underline{U}_i(t)$ $(1 \leq i \leq n)$. We want to find $\underline{M}$ and $\underline{B}^*$ such that these flows can be attained in

the finite interval $[0, T]$ and overall cost is a minimum. In short, the problem may be stated as: find $\underline{B}^*$ and $\underline{M}$ subject to constraints (4) through (6) for all t in $[0, T]$ such that

$$\underline{C}^t (\underline{B}^* - \underline{B}^*_p) + \underline{K}^t (\underline{M} - \underline{M}_p) \tag{9}$$

is minimized. Here we must add two additional side conditions $\underline{B}^* \geq \underline{B}^*_p$ and $\underline{M} \geq \underline{M}_p$. Vectors $\underline{C}$ and $\underline{K}$ are defined as in Problem 1. The problem again is solvable if and only if the given specifications satisfy (8).

Problem 3. Evaluation of the Network Traffic Handling Capability: We would like to know what is the maximum flow handling capacity of a network. A way to formulate the problem is as follows: We assume we are given a network N, thus a graph G, and its capacity vectors $\underline{B}^*$ and $\underline{M}$ and initial state vectors $\underline{R}_1, \underline{R}_2, \ldots, \underline{R}_n$. We are also given n sources of messages, but the corresponding sink vectors are not completely specified. We would like to maximize

$$\sum_{i=1}^{m} d_i \sum_{j=1}^{m} e_j \int_o^T u_{ij}(t)\, dt \tag{10}$$

(e_j's and d_i's are 0's or 1's and they depend upon the choice of the desired subset of flows to be maximized and $u_{ij}(t)$ is the jth component of $\underline{U}_i(t)$) subject to the constraints (4), (5), and (6) and the constraint expressed by (8). We might like to add constraints such as $\underline{U}_i(t) \geq \underline{U}_{io}$ for $i = 1,2,\ldots, n$ and all t in the interval $[0, T]$ where $\underline{U}_{io}$ are given fixed vectors. However, addition of such constraints may make the problem unsolvable; that is the flow requirements may not be feasible (or attainable) [7].

III. Computational Aspects

To solve any of the three problems discussed in the previous section by a digital computer, we must quantize time. So, we assume

t is a discrete variable and we replace all of the integrals by their equivalent summations. Then, all constraints and objective functions become clearly linear algebraic equations and thus they are all solvable by linear programming methods. The number of constraints in each problem in a practical situation may be exceedingly large. However, the constraints have a special structure, which makes it possible to use the Dantzig-Wolfe decomposition principle [11, 12]; thus we can break the large linear program into many smaller ones. With this idea in mind, it is the feeling of the author that up to 10^4 constraints can be handled with the present day digital computers. There are a number of questions that arise in connection with the design problem.

(i) It is clear that the choice of "linear cost," as implied by (7), is not a practical one but was made to simplify the computation. A more practical choice is to pick the cost of b units of capacity (be it memory or channel capacity) as a fixed cost plus a linear cost proportional to b. To do this, we replace objective function (7) by

$$C(\underline{B}^*) + K(\underline{M}) \tag{11}$$

and the functions $C(\underline{B}^*)$ and $K(\underline{M})$ are scalar valued functions of vectors $\underline{B}^*$ and $\underline{M}$ and are defined by

$$C(\underline{B}^*) = \sum_{i=1}^{m} c_i\,(b_i^*) \text{ and } K(\underline{M}) = \sum_{i=1}^{n} k_i(m_i) \tag{12}$$

where the functions $c_i(b_i^*)$ and $k_i(m_i)$ are defined as

$$c_i(b_i^*) = \begin{cases} c_i b_i^* + p_i\,, & \text{if } b_i^* > 0 \\ 0\,, & \text{if } b_i^* = 0 \end{cases} \tag{13}$$

$$k_i(m_i) = \begin{cases} k_i m_i + q_i\,, & \text{if } m_i > 0 \\ 0\,, & \text{if } m_i = 0 \end{cases} \tag{14}$$

and p_i and q_i are initial costs of branch b_i and the memory unit m_i. Such a generalization although theoretically possible computationally is substantially more difficult [13,14] .

(ii) How does one choose the original topology or structure of the network? The choice may partially be made by some engineering considerations. But it is also conceivable that one might base the formulation on a complete graph (a graph in which there is a branch connecting every pair of distinct vertices). Then, using (11) as an objective function, it is likely that capacities of many branches and memory units will be zero. In any case, an optimum choice will be made. One might alternately try the original objective function (7) and solve the problem, and then, set to zero the capacities of those branches and memory units which are small and repeat the problem again. This procedure may not lead to the optimum results, but it will certainly be better than an arbitrary choice.

(iii) How does one choose the initial state vectors $\underline{R}_1, \underline{R}_2, \ldots, \underline{R}_n$? This question is related to the choice of length of period T in a design problem. In the sense that if one is willing to take T to be sufficiently large, it is perfectly acceptable to set $\underline{R}_i = \underline{0}$ for $i = 1, 2, \ldots, n$. Most sources of traffic have a periodic nature. It is reasonable to assume that the content of memory reaches a "steady state" after the lapse of several periods. Once this "steady state" is reached, then we want to make sure that the designed network is capable of handling the traffic that is forced into it. Gale [15] and later Frank [16] study a similar problem in great details in connection with transient flows in communication networks with delay, but no memory. (See the discussion at the end of this section).

(iv) What does one do if the communication network is not unilateral in nature? This question fortunately can be answered easily. One can

replace each branch b_i in G whose capacity is b_i^* by two parallel branches b_{i1} and b_{i2} in G_1 with arrowheads placed on them in opposite directions. Then consider resulting directed graph as a model of our communication network. If we let the capacity of b_{i1} = b_{i1}^* and capacity of b_{i2} = b_{i2}^*, then in each of the problems discussed in Section II, one must add another constraint

$$b_{i1}^* + b_{i2}^* \leq b_i^* \quad , \quad i = 1, 2, \ldots, m \tag{15}$$

Because of the computational difficulties one encounters in the solution of large linear programs, it is often valuable to have a feasible non-optimum solution [11] . Here, we will give a procedure for the design problem (Problem 1) which leads to a "sub-optimal" solution without any serious computational difficulties. For the sake of the present discussion, we assume that the messages from each source lead to one destination; thus, one might have up to n sources connected to the same vertex.

Let $s_i^j(t)$ be the rate of the flow of the messages which are destined for vertex v_j and are produced by a source which is connected to vertex v_i $(i,j = 1,2,\ldots, n)$. Such a source will be called source s_i^j. Let $s_i^j(t)$ be a periodic function of period T, and let

$$a_i^j = \frac{1}{T}\int_0^T s_i^j(t)\,dt \qquad \text{for } i,j = 1,2,\ldots, n. \tag{16}$$

Let c_i^j be the cost per unit capacity of link (path) between vertices v_i and v_j $(1 \leq i,j \leq n)$. We assume $c_i^i = 0$, and $c_i^j, (i \neq j,\ i,j = 1,2,\ldots,n)$, is computed by finding the shortest path between vertices v_i and v_j in graph G when the weight of branch b_k is equal to $c_k, (k = 1,2,\ldots,m)$. Let k_i be the cost per unit capacity of the memory unit m_i. The cost of providing sufficient link (channel) capacity to be able to

directly forward the messages produced by the source s_i^j is

$$L_{ij} = c_i^j \left[\sup_{0 \le t \le T} s_i^j(t) \right] \quad \text{for } i,j = 1,2, \ldots, n. \tag{17}$$

On the other hand, if we use store and forward method of transmission, it will cost no more than

$$K_{ij} = c_i^j \; a_i^j + k_i \left\{ r_i^j(0) + \sup_{0 \le t \le T} \int_o^t \left[s_i^j(\tau) - a_i^j \right] d\tau \right\}$$

for $i,j = 1, 2, \ldots, n.$ (18)

Thus, for the source s_i^j, a decision in favor of direct traffic is made if

$$\min \left[L_{ij}, K_{ij} \right] = L_{ij};\ i,\ j = 1,\ 2,\ \ldots,\ n \tag{19}$$

otherwise, the store-forward method is preferred. Clearly this method can be carried out for each source separately and the overall cost of the network is

$$\sum_{i,j=1}^{n} \min \left[L_{ij}, K_{ij} \right] \tag{20}$$

It should be noted that the second term in (18) is identified with the cost of memory requirement for the source s_i^j. Thus, we may write

$$m_i^j = r_i^j(0) + \sup_{0 \le t \le T} \int_o^t \left[s_i^j(\tau) - a_i^j \right] d\tau \tag{21}$$

Where $r_i^j(0)$ is the initial content of that memory unit. Due to the choice of output flow a_i^j of the memory unit, $r_i^j(0) = r_i^j(T)$, thus the content of the memory is in a "steady-state" condition regardless of its value. However, in a design problem one is not usually given the initial content $r_i^j(0)$ of the memory and it is up to the designer to make a judicious choice. To do this, we must remember that $r_i^j(0)$ must be chosen such that the following condition is satisfied for all t in $[0,T]$.

$$r_i^j(0) + \int_o^t \left[s_i^j(\tau) - a_i^j \right] d\tau \geq 0 \tag{22}$$

Since the smaller $r_i^j(0)$ the lesser the cost of the memory unit, the best choice for $r_i^j(0)$ is

$$r_i^j(0) = - \inf_{0 \le t \le T} \int_0^t [s_i^j(\tau) - a_i^j] \, d\tau \tag{23}$$

which is a positive number because a_i^j is the average value of $s_i^j(t)$.

IV. Design Considerations of Memory Systems

In practical situations, the messages flowing through the network may have different priorities. Let us assume, we have h classes of messages, where messages in class one have the highest priority and the messages in class two have the next highest priority and so forth. Since the transit time through the channels is assumed to be negligible, we may require that the messages as they arrive in a memory unit should be processed in the order of their priorities. Two possible ways of designing a memory unit to handle such traffic comes to mind: "parallel" or "series". Typical examples of such designs are shown in Figs. 1.2a and 1.2b. We shall first discuss the parallel model and then extend the results to the series case.

Assume that the messages arrive at vertex v_i at the rate $x_i(t)$, and that $x_{i1}(t)$, $x_{i2}(t)$, ..., $x_{ih}(t)$ are the rates of the first, second ... and hth priority classes. We have, $\sum_{k=1}^{n} x_{ik}(t) = x_i(t)$. The kth priority messages are stored in the memory unit designated by (the box) m_{ik} and they leave that box at the rate of $y_{ik}(t)$ $(1 \le k \le h)$. We also have $\sum_{k=1}^{h} y_{ik}(t) = y_i(t)$. As before, we assume that m_{ik} also represent the capacity of the memory unit m_{ik}, $(1 \le i \le n,\ 1 \le k \le h)$. Let t_{ik} be a given number which represents the maximum length of time that a message stored in m_{ik} is allowed to remain there $(1 \le i \le n,\ 1 \le k \le h)$. (One expects that $t_{ik} > t_{i,k+1}$, for $i = 1, 2, ..., n$ and $k = 1, 2, ..., h-1$). Let $r_{ik}(t)$ represent

the messa[illegible] content of m_{ik} at time t. To ensure that these messages will not remain in m_{ik} longer than t_{ik} units of time, we must have

$$r_{ik}(t) \le \int_{t}^{t+t_{ik}} y_{ik}(\tau)\, d\tau \quad \text{for } i = 1,2, \ldots n \; ; \; k = 1,2,\ldots h \text{ and all } t \tag{24}$$

We will now attempt a general formulation for the design of communication networks with memory and different priority messages. We need to define the notation.

Let $s_{ij}(t)$ be the output rate of jth priority messages from the source connected to v_i at time t and let $\underline{U}_{ij}(t)$ be the sink vector of these messages at time t. Let the vector $\underline{Z}_{ij}(t)$ represent the rates of flows at time t in the branches of G (or G^*) due to the jth priority messages originating from the source s_i. Let the vectors $\underline{X}_{ij}(t)$ (and $\underline{Y}_{ij}(t)$) be vectors whose kth component represents the rate of flow into (and out of) memory unit m_{kj} due to the jth priority messages originating from the source s_i. Let the state vector $\underline{R}_{ij}$ be a vector whose kth component represent the content of the memory unit m_{kj} due to jth priority messages originating from s_i at the time $t = 0$. Let column vector $\underline{M}_j$ be defined as

$$\underline{M}_j = (m_{1j}\; m_{2j} \;\ldots\; m_{nj})' .$$

Then, we may write for each t in a given interval $[0,T]$

$$A_1 \begin{bmatrix} \underline{Z}_{ij}(t) \\ \\ s_{ij}(t) \end{bmatrix} = \underline{X}_{ij}(t) \quad \text{and} \quad A_2 \begin{bmatrix} \underline{Z}_{ij}(t) \\ \\ \underline{U}_{ij}(t) \end{bmatrix} = \underline{Y}_{ij}(t) \tag{25}$$

for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, h$

We must have the branch capacity constraint

$$\sum_{i=1}^{n} \sum_{j=1}^{h} \underline{Z}_{ij}(t) \leq B^{*} \quad \text{and} \quad \underline{Z}_{ij}(t) \geq \underline{0} \tag{26}$$

for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, h$

and memory capacity constraints

$$\sum_{i=1}^{n} \underline{R}_{ij} + \int_{0}^{t} \sum_{i=1}^{n} [\underline{X}_{ij}(\tau) - \underline{Y}_{ij}(\tau)]\, d\tau \leq \underline{M}_{j}$$

$j = 1, 2, \ldots, h$

and

$$\underline{R}_{ij} + \int_{0}^{t} [\underline{X}_{ij}(\tau) - \underline{Y}_{ij}(\tau)]\, d\tau \geq \underline{0} \tag{27}$$

for $i = 1, 2, \ldots n$ and $j = 1, 2, \ldots h$

and the maximum allowable delay constraint (23), which might be rewritten as

$$r_{ik} + \int_{0}^{t} [x_{ik}(\tau) - y_{ik}(\tau)]d\tau \leq \int_{t}^{t+t_{ik}} y_{ik}(\tau)\, d\tau \tag{28}$$

for $i = 1, 2, \ldots n$ and $k = 1, 2, \ldots h$

where r_{ik}, $x_{ik}(t)$, and $y_{ik}(t)$ are the ith components of vectors

$$\sum_{j=1}^{n} \underline{R}_{jk}, \quad \sum_{j=1}^{n} \underline{X}_{jk}(t), \text{ and } \sum_{j=1}^{n} \underline{Y}_{jk}(t), \text{ respectively.}$$

The objective function in the case of the design (linear cost) is to

$$\text{minimize } \underline{C}^{t}\underline{B}^{*} + \sum_{j=1}^{h} \underline{K}_{j}^{t}\, \underline{M}_{j} \tag{29}$$

where the jth component $\underline{K}_j$ is the cost per unit capacity of memory unit m_{ij} and as before jth component of $\underline{C}$ is the cost per unit capacity of branch b_i.

It is possible to reformulate the communication network with h levels of priorities based on the series memory model (see, Fig. 2b). To do this, let r_{kj} be the content of memory unit m_{kj} at $t = 0$ and let $x_{kj}(t)$ and $y^*_{kj}(t)$ be the input and the output flow rates of the memory unit m_{kj} at time t. Constraints (25) and (26) remain intact and we must replace (27) and (28) by

$$0 \le r_{kj} + \int_0^t [x_{kj}(\tau) + y^*_{k,j+1}(\tau) - y^*_{kj}(\tau)]\, d\tau \le m_{kj} \tag{30}$$

$$k = 1, 2, \ldots, n;\ j = 1, 2, \ldots, h \text{ and all } t$$

and

$$r_{kj} + \int_0^t x_{kj}(\tau) + y^*_{kj+1}(\tau) - y^*_{kj}(\tau)\, d\tau \le \int_t^{t+t^*_{kj}} y^*_{kj}(\tau)\, d\tau$$

$$k = 1,2, \ldots, n;\ j = 1,2, \ldots, h \text{ and all } t \tag{31}$$

where $y^*_{kh+1}(t) = 0$. We may compute t^*_{kj} from the given t_{kj} recursively as follows

$$t^*_{kj} = t_{kj} - \sum_{i=0}^{j-1} t^*_{ki}\ , \quad k = 1, 2, \ldots, n;\quad j = 1, 2, \ldots, h \tag{32}$$

where $t^*_{ko} = 0$

In addition we must add that for each j $(1 \le j \le n)$

$$x_{kj}(t) = \text{kth component of } \sum_{i=1}^{n} \underline{X}_{ij}(t) \text{ for } k = 1, 2, \ldots, n. \tag{33}$$

and

$$y^*_{k1}(t) = \text{kth component of } \sum_{i=1}^{n} \sum_{j=1}^{h} \underline{Y}_{ij}(t) \tag{34}$$

The objective function will be to

$$\text{minimize } \underline{C}^t\, \underline{B}^* + \sum_{i=1}^{n} \sum_{j=1}^{h} k_{ij}\, m_{ij} \tag{35}$$

To illustrate the number of constraints involved in these formulations, suppose m is the number of branches, n is the number of vertices, h is the number of priority levels, and T is the time that the communication network is under consideration. If we assume that t assumes values at integral units of time, then the number of constraints for the formulation with parallel memory (Fig. 2a) is given by

$$3n^2hT + mT + nhmT + 2nhT \tag{36}$$

which could be an extremely large number.*

V. An Approximate Model of Highway or Street Traffic

To develop an accurate model and a general formulation of street traffic is extremely difficult [17]. In this section, we will present an approximate formulation of street traffic leading to a large linear program. Let a directed graph G represent a highway or transportation network N. Let the branches and vertices of G represent streets (or highways) and street intersections (or junctions) of N, respectively. If a street (or highway) can handle traffic in both directions, then this is represented in G by two directed branches in "parallel", (i.e., two branches connected between the same pair of vertices), whose arrowheads are in opposite directions. Network N could be considered to be ten square city blocks, or a highway transportation network interconnecting 100 cities. The effect of remaining part of this network is reflected into network N through the n sources and sinks of traffic connected to N (or graph G). Let G have m branches and n vertices. As before, we assume we have n sources of traffic whose output rates as functions of time are $s_1(t)$, $s_2(t)$, ..., $s_n(t)$

---

* In this formulation, we must be given $y_{kj}(t)$ for $0 \le t \le t_{kj}$, $(1 \le k \le n,\ 1 \le j \le h)$

which are connected to the vertices $v_1, v_2, \ldots, v_n$ of G. Corresponding to each source s_i there is sink column n-vector $\underline{U}_i(t) = [u_1^i(t) \;\; u_2^i(t) \;\ldots\; u_n^i(t)]'$ where $u_k^i(t)$ is the rate of arrival of traffic at vertex v_k, due to traffic originating from the source s_i, as a function of time. Let us assume that it takes τ_j units of time for a car to travel from an end of (tail of the arrow) branch b_j to the other end (head of the arrow) $(1 \le j \le m)$. Let $x_j^i(t)$ represent the rate of traffic at time t at the tail end of the branch b_j due to the source s_i. Let $z_j^i(t)$ be the traffic at the head of branch b_j at time t due to the source s_i (see, Fig. 3). We will assume

$$z_j^i(t) = x_j^i(t + \tau_j) \text{ for } i = 1, 2, \ldots, n; \;\; j = 1, 2, \ldots, m \text{ and all } t \qquad (37)$$

At the "head end" of branch b_j, the traffic flow in b_j enters a junction or vertex. Since at a junction, one might have a stop sign (or stop light), we assume the traffic flow $z_j^i(t)$ itself does not necessarily flow into this junction. But $z_j^i(t)$ enters a holding box whose output $y_j^i(t)$ enters the intersection (or junction) and whose initial content doe to s_i is $r_j^i(0)$. This assumption will lead to the following equation

$$r_j^i(0) + \int_0^t [z_j^i(\tau) - y_j^i(\tau)]\, d\tau \ge 0 \text{ for } i = 1,2,\ldots, n; \; j = 1,2,\ldots,m \text{ and all } t. \qquad (38)$$

Since the traffic may not be allowed to back up beyond certain limits, we have

$$\sum_{i=1}^{n} r_j^i(0) + \int_0^t [\sum_{i=1}^{n} z_j^i(\tau) - \sum_{i=1}^{n} y_j^i(\tau)]\, d\tau \le M_j$$
$$j = 1, 2, \ldots, m \text{ and all } t. \qquad (39)$$

Assuming branch b_j has capacity b_j^*, we have a constraint

$$\sum_{i=1}^{n} x_j^i(t) \leq b_j^* \quad \text{for } j = 1, 2, \ldots, m \text{ and all } t \tag{40}$$

and we also have

$$x_j^i(t) \geq 0 \text{ and } y_j^i(t) \geq 0 \quad \text{for } i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m \text{ and all } t \tag{41}$$

It should be noted that in a physical highway traffic problem, $y_j^i(t)$ is zero at certain intervals of time because the traffic lights are periodically red. This type of consideration, would make the problem substantially more difficult. Here, we assume $y_j^i(t)$ represents an average value taken over one or more cycles of the traffic light.

The next step in our formulation is to consider the traffic flow at each junction. We first define for each vertex v_j, $(j = 1,2,\ldots, n)$, two sets of integers:

$$\alpha_j = \left\{ k \;\middle|\; b_k \text{ is incident at } v_j \text{ with arrowhead toward } v_j \right\}$$

$$\beta_j = \left\{ \ell \;\middle|\; b_\ell \text{ is incident at } v_j \text{ with arrowheads away from } v_j \right\}$$

By $y_p^i(t)$, $p \epsilon \alpha_j$, we mean the traffic flow entering vertex (or junction) v_j on branch b_p due to the source s_i. Let $y_{pq}^i(t)$, $p \in \alpha_j$ and $q \in \beta_j$, be that portion of $y_p^i(t)$ which enters the branch b_q. Thus we have

$$u_{jp}^i(t) + \sum_{q \in \beta_j} y_{pq}^i(t) = y_p^i(t), \; p \epsilon \alpha_j, \text{ and } i,j = 1, 2, \ldots, n \text{ and all } t \tag{42}$$

where the summation is over all integers in β_j and $u_{jp}^i(t)$ represent that portion of $y_p^i(t)$ that flows into the sink at vertex v_j.

We must also add

$$\begin{cases} \displaystyle\sum_{p \in \alpha_j} y_{pq}^i(t) = x_q^i(t), \text{ for } q \epsilon \beta_j,\ i,j = 1, 2, \ldots, n, \text{ and } i \neq j \\ \displaystyle\sum_{p \epsilon \alpha_j} y_{pq}^i(t) + s_i^q(t) = x_q^i(t), \text{ for } q \epsilon \beta_j,\ i = j = 1, 2, \ldots, n \end{cases} \tag{44}$$

where $s_i^q(t)$ is the portion of $s_i(t)$ which is entering branch b_q $(q\epsilon\beta_j)$ and,

$$\sum_{q\epsilon\beta_j} s_i^q(t) = s_i(t) \quad i = 1, 2, \ldots, n \text{ and all } t. \tag{45}$$

To continue with our formulation of flows at a junction, we must introduce a number of inequality constraints as follows:

$$s_i^q(t) \geq 0 \text{ , } i = 1, 2, \ldots, n; \quad q\epsilon\beta_j \text{ and all } t \tag{46}$$

$$y_{pq}^i(t) \geq 0 \text{ , } i = 1, 2, \ldots, n; \quad p\epsilon\alpha_j; \; q\epsilon\beta_j; \; j = 1, 2, \ldots, n \text{ and all } t \tag{47}$$

$$u_{jp}^i(t) \geq 0, \; i = 1, 2, \ldots, n; \; p\epsilon\alpha_j \; ; \; j = 1, 2, \ldots, n \text{ and all } t. \tag{48}$$

It is difficult to establish upper bounds for the rate of traffic flow through a junction. Roughly speaking, the traffic flow at an intersection is a function of the physical size of the intersection, the cycle of the traffic light, and the number of cars that are making turns at the intersection thus disrupting the smooth flow of traffic. (Generally left turns cause the greatest difficulty. Note that, in our present model, we have allowed even a "U turn" at an intersection). Before we formulate upper bound constraints for the flow through a typical vertex (or junction), v_j, we will define a few terms. Let

$$\sum_{i=1}^{n} y_{pq}^i(t) = y_{pq}(t), \; p\epsilon\beta_j \; ; \; j = 1, 2, \ldots, n. \tag{49}$$

We define a weighing factor (a non-negative number) for every portion of the traffic through a junction. Let W_{pq} be the weighing factor corresponding to $y_{pq}(t)$. $(p\epsilon\alpha_j \; ; \; q\epsilon\beta_j)$. For example, one might set $W_{pq} = 1$, if branches b_p and b_q corresponded to the same streets, (i.e. the traffic corresponding to $y_{pq}(t)$ was moving in a straight path), and one might set $W_{pq} = 2$ if $y_{pq}(t)$ corresponds to traffic

which has to make a turn to enter branch (street) b_q. With these ideas in mind, it is reasonable to add the following constraint

$$\sum_{q\epsilon\beta_j} W_{pq}\, y_{pq}(t) \le W_p\,,$$

$$p\epsilon\alpha_j\ ;\ j = 1, 2, \ldots, n \text{ and all } t \qquad (50)$$

The accuracy of our formulation to a great extent depends on the accuracy of the above constraint (50). However, this is intended to be an approximate formulation of a very difficult problem.

To finish our formulation, we need to specify an objective function. To do this, however, we need to describe exactly what is the problem. As in the case of the communication network with memory, there seems to be three classes of problems in connection with traffic networks: (1) analysis of the existing capability, (2) design of an entirely new traffic network, and (3) expansion of the present capability.

(1) Analysis: We assume n sources of traffic are given for all t in [0, T], but the corresponding sink vectors are not completely specified. The structure of the network is given in the form of graph G, and we are given constants b_j^*, τ_j, M_j, W_{pq}, W_p, ($p\epsilon\alpha_i$ and $q\epsilon\beta_i$), and $r_j^i(0)$, for $i = 1, 2, \ldots, n$ and for $j = 1, 2, \ldots, m$.*
Then, we would like to maximize

$$\sum_{i=1}^{n} d_i \sum_{j=1}^{n} e_j^{\prime} \int_0^T u_i^j(t)\ dt \qquad (51)$$

(d_i's and e_j's are 0's or 1's and they depend upon the choice of the desired subset of flows to be maximized and u_i^j is the ith component

---

* In all of these problems, because of the branch delays, we must also be given, $z_j^i(t)$ for $0 \le t \le \tau_j$ $(1 \le j \le m,\ 1 \le i \le n)$.

of vector $\underline{U}_j(t)$) subject to constraints (37) through (50) being satisfied for all t in [0, T], and in addition the following constraint must be satisfied.

$$\sum_{j=1}^{n} r_j^i(0) + \int_0^T s_i(t) \geq \sum_{j=1}^{m} \int_0^T u_j^i(t)\, dt \text{ for } i = 1, 2, \ldots, n \quad (52)$$

This problem is not quite as practival as the corresponding problem in communication network; because one is not free to direct traffic as one pleases, i.e., the driver usually picks the shortest route to reach his destination and he is not so concerned about the efficient use of the network.

(2) Design: We are given G, $s_i(t)$, $\underline{U}_i(t)$, M_j, τ_j, $r_j^i(0)$, W_{pq}, ($p\epsilon\alpha_i$, $q\epsilon\beta_i$; $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$, t in $[0,T]$), and also given cost per unit capacity of the links $c_i (i = 1, 2, \ldots n)$ and the cost of junctions $k_p (p\epsilon\alpha_j, j = 1, 2, \ldots, n)$. Then we would like to minimize

$$\sum_{i=1}^{m} c_i b_i^* + \sum_{j=1}^{n} \sum_{p\epsilon\alpha_j} k_p W_p \quad (53)$$

subject to constraints (37) through (50) being satisfied for all t in [0,T].

(3) Expansion: Given are all of the quantities in (2) and also present capacities $b_i^{*'}$, ($i = 1, 2, \ldots, m$), and W'_p, ($p\epsilon\alpha_j$, $j = 1, 2, \ldots, n$), We would like to find $b_i^* \geq b_i^{*'}$ and $W_p \geq W'_p$ such that constraints (37) through (50) are satisfied for all t in [0,T] and

$$\sum_{i=1}^{m} c_i (b_i^* - b_i^{*'}) + \sum_{j=1}^{n} \sum_{p\epsilon\alpha_j} k_p (W_p - W'_p) \quad (54)$$

is minimized.

VI Conclusion

There are two fundamental problems with respect to the linear programming formulation of communication networks with memory (as well as the traffic networks): (1) The enormous number of constraints which to say the least limits the size of the networks that can be handled even with the fastest available digital computers. Some aspects of this problem were discussed in Section III. As it can be seen, computation problem will increase substantially when one puts limitations on acceptable limits on delay of message at each storage point, (as discussed in Section IV). This problem becomes even worse, when there are a number of priority classes in the messages that are flowing through the network. But these computational difficulties are to great extent inherent in the problem and, of course, the choice of the model. (2) The second problem is the choice of a deterministic model. It is well known that the nature of such traffic and sources of messages (or traffic) is statistical. The question is "can anything be done with a statistical model?" For a communication network with direct traffic (no memory) and single commodity traffic, Frank and Hakimi [18,19] have presented a statistical analysis which lead to grave computational difficulties. This rules out the possibility of the generalization of their results to the present case, as a practical solution. Kleinrock [20] presented a statistical analysis of message flow and delay in a direct traffic communication network where the delay time was due to the traveling time in the branches. Kleinrock's results, although interesting, is very difficult to apply to the present problem and also leads to serious computational difficulties. However, since the memory units have a smoothing effect on the traffic through the communication network, it is more reasonable to accept

a deterministic model for the present problem than it is to accept a deterministic model for communication networks without memory.

It might be convenient to use the following approach in the sub-optimum design of a communication network with memory. Let us assume we are given the initial conditions and a period T. The network N is to be designed to handle the given flow specifications over the internal $[0,T]$. Let $T_1 < T$ and suppose we first design the network N_1 which is capable of handling the flow specifications over the interval $[0,T_1]$. Then, using the contents of the memory units at time T_1 as the new initial conditions, we modify network N_1 (in a optimum manner as in Problem 2) such that the resulting network N can handle the flow specifications over the interval $[T_1,T]$. Repeated use of this idea can help reduce the computational time. The same idea can be used (with the appropriate choice of initial conditions) in connection with the street traffic problem and the communication networks with message priorities. Unfortunately, the resulting realization is not optimum.

Acknowledgement

The author wishes to acknowledge the helpful discussion held with Dr. W. H. Kim of Columbia University. I am also indebted to Messrs. R. G. DeWitt, R. L. Schmal, O. G. Becker, and especially L. Stier of Western Union who suggested the problem and its ramifications to the author. I am also grateful to Prof. Z. V. Rekasius of Northwestern University whose basic suggestion helped to formulate the case of messages with different priorities.

This work was, in part, performed for Faculties Associated Consultants, Inc. with the support of Western Union Telegraph Company.

REFERENCES

[1]. J. Otterman, "Grade of Service of Direct Traffic Mixed with Store and Foreard Traffic", IRE Trans. on Communication Systems, Vol CS-10, pp. 130-138, March, 1962.

[2]. H. E. Brooks, "Circuit Switching Network for an Advanced Record Systems", Western Union Technical Review, pp. 70-76, April, 1966

[3]. D. E. Carruth, "ARS, Advanced Record Systems", Western Union Technical Review, pp. 8-17, Jan., 1966

[4]. L. R. Ford, Jr. and D. R. Fulkerson, "Maximum Flow Through a Communication Network" Can. J. Math, vol 8, pp. 399-404, 1956

[5]. W. Mayeda, "Terminal and Branch Capacity Matrices of a Communication Net", IRE Trans. on Circuit Theory, vol CT-7, pp. 261-269, Sept., 1960

[6]. O. Wing and R. T. Chien, "Optimal Synthesis of Communication Nets", IRE Trans. on Circuit Theory, vol CT-8, pp. 44-49, March, 1961

[7]. S. L. Hakimi, "Simultaneous Flows through a Communication Network", IRE Trans. on Circuit Theory, vol CT-9, pp. 169-175, June, 1962.

[8]. N. Deo and S. L. Hakimi, "Minimum Cut Increase of the Terminal Capacities of a Communication Network", IEEE Trans. on Communication Tech., vol. com-14, pp. 63-64, Feb., 1966

[9]. L. R. Ford, Jr. and D. R. Fulkerson, Flows in Networks, Princeton Univ. Press, Princeton, N.J., 1962

[10]. J. A. Tomlin, "minimum Cost Multicommodity Network Flows", Operation Research, vol. 14, pp. 45-51, Jan-Feb, 1966

[11]. G. B. Dantzig, Linear Programming and Extensions, Princeton Univ. Press, Princeton, N.J., 1963

[12]. G. B. Dantzig and P. Wolfe, "The Decomposition Algorithm for Linear Programs" Econometrica, vol. 29, pp. 767-778, 1961

[13]. W. M. Hirsch and G. B. Dantzig, "The Fixed Charge Problem", the RAND Corp. Paper, p-648, Dec. 1954.

[14]. W. M. Hirsch and A. J. Hoffman, "Extreme Varieties, Concave Functions and the Fixed Charge Problem", Communications on Pure and Applied Math, vol 19, pp. 355-369, Aug., 1961

[15]. D. Gale, "Transient Flows in Networks", Mich. Math. J., vol. 6, pp. 59-63, 1958.

[16]. H. Frank, "Dynamic Communication Networks" (Unpublished paper), Dept. of Elect. Eng'rg., Univ. of Calif., Berkeley, Calif., July, 1966.

[17]. A. Charnes and W. W. Cooper, "Management Models and Industrial Application of Linear Programming, vol. II, pp. 785-798, John Wiley and Sons, New York, N.Y., 1961

[18]. H. Frank and S. L. Hakimi, "Probabilistic Flows through a Communication Network", IEEE Trans. on Circuit Theory, vol. CT-12, pp. 413-414, Sept., 1965

[19]. H. Frank and S. L. Hakimi, "On the Optimum Synthesis of Statistical Communication Nets", unpublished paper, Dept. of Elect. Enging'rg, Northwestern Univ., Evanston, Ill., June, 1968.

[20]. L. Kleinrock, Communication Nets-Statistical Message Flow and Delay, McGraw Hill Book Co., New York; 1964

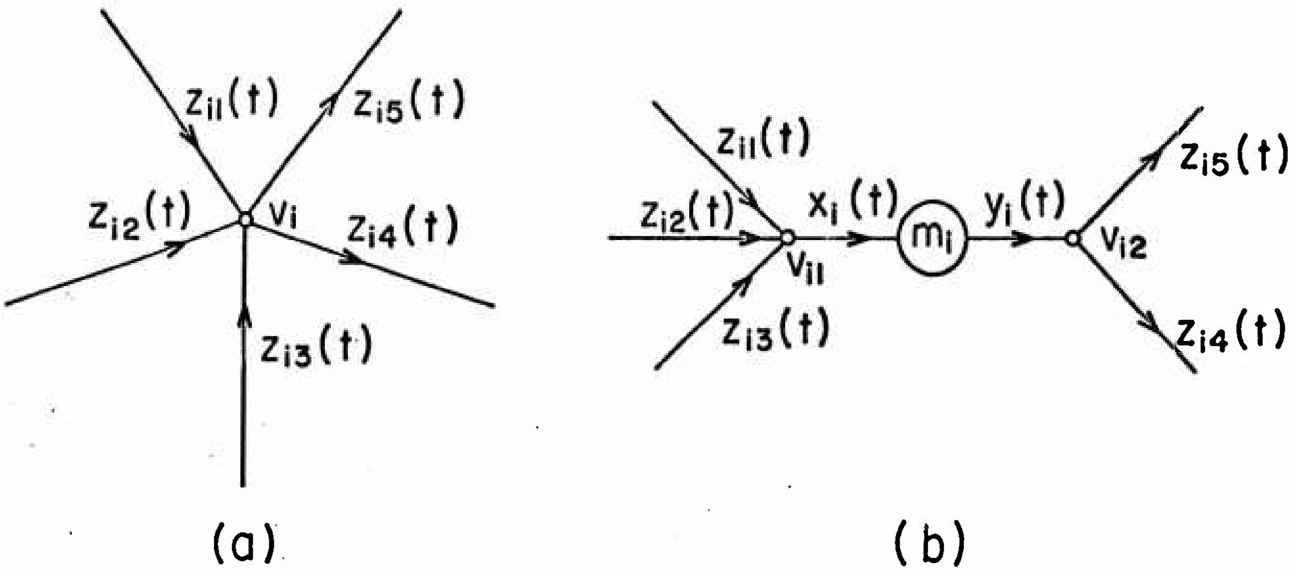


Fig. 1 - (a) Vertex v_i , and (b) splitting of v_i and introduction of memory unit.

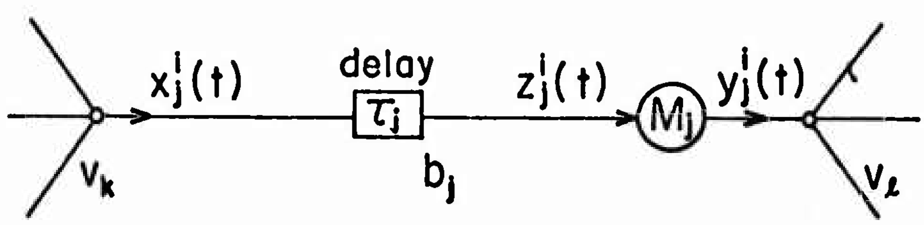
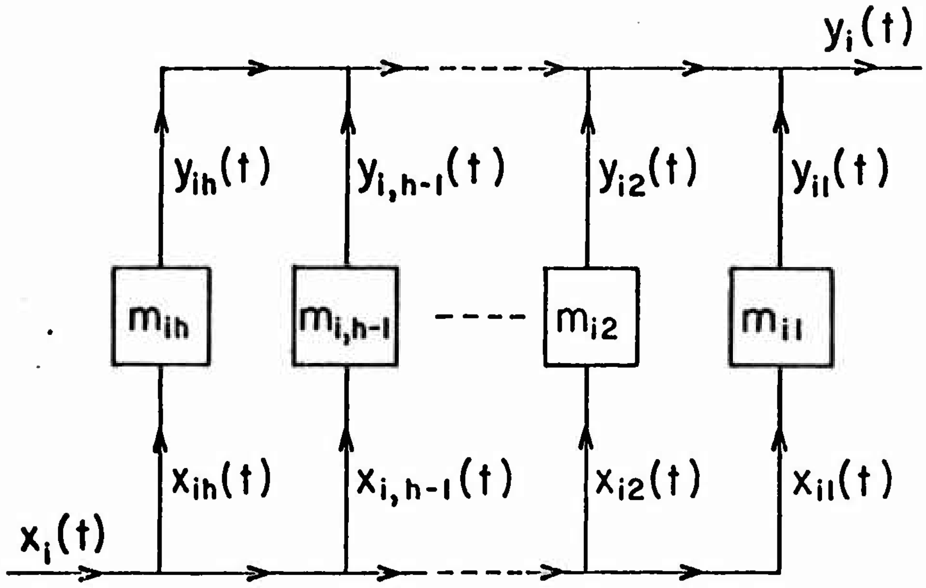
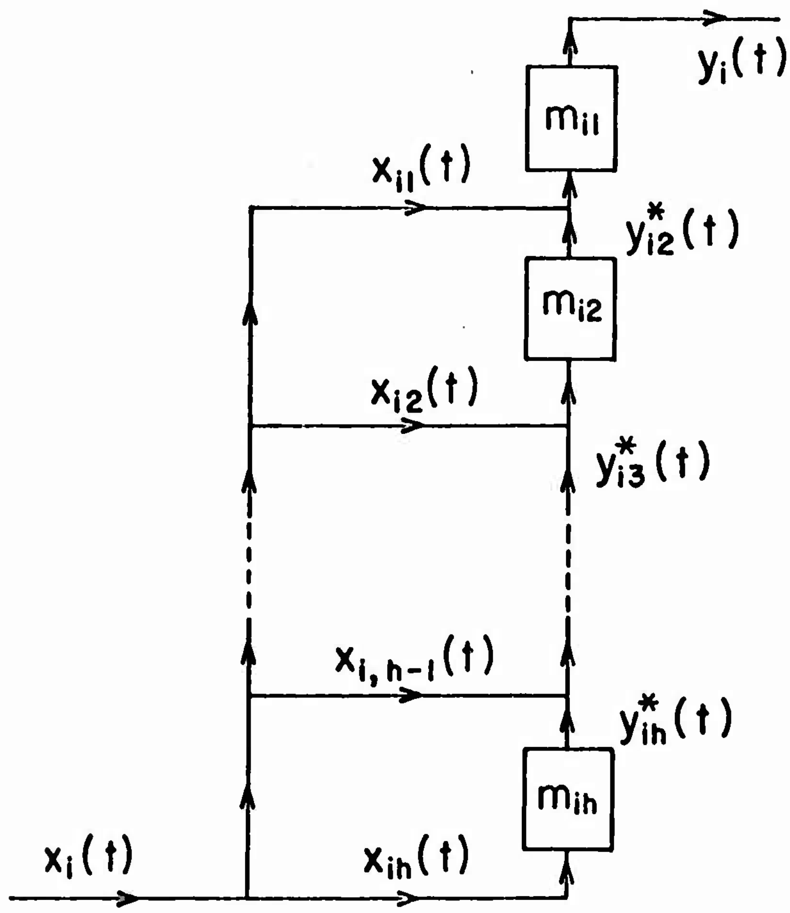


Fig. 3 - Representation of single link in a highway network.



(a)



(b)

Fig. 2 - Typical memory systems, (a) parallel, (b) series.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Information-Processing And Control SYSTEMS LABORATORY Northwestern University, Evanton, Illinois 60201	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b GROUP Not applicable

3. REPORT TITLE

ANALYSIS AND DESIGN OF COMMUNICATION NETWORKS WITH MEMORY

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Not applicable

5. AUTHOR(S) *(Last name, first name, initial)*

Hakimi, S. Louis

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
January 1967	32	20

8a. CONTRACT XXXXXXX NO. N00014-66-C0020-A03 Grant No. AF-AFOSR-98-65 b. PROJECT NO. NR 373-502/3-14-66 c. Not applicable d. Not applicable	9a. ORIGINATOR'S REPORT NUMBER(S) 67-103 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* Not applicable

10. AVAILABILITY/LIMITATION NOTICES

Qualified requesters may obtain copies of this report from the Defense Documentation Center.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
Not applicable	U. S. Air Force Office of Scientific Research, Washington, D. C. and Office of Naval Research, Washington, D. C.

13. ABSTRACT

A mathematical formulation of the communication networks with memory is presented assuming that the sources of traffic are deterministic but not necessarily time invariant. The formulation leads to a linear programming problem. Some generalizations and justifications of the choice of the model are discussed. The same basic formulation can be used as a tool for analysis as well as least-cost design or improvement of an existing network. Design of the memory systems and its relation with messages with priorities is considered.
Similar concepts are used to arrive at an approximate linear programming formulation of "street traffic".

Unclassified

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Communication networks						
Traffic networks						
Delay and Memory in Communication Networks						
Linear Programming						
Optimum Design, Analysis, Optimum improvement of communication of Communication Networks with Memory						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through

_________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through

_________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through

_________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

Unclassified

Security Classification